

Remarks

Reconsideration of the application is respectfully requested in view of the foregoing amendments and following remarks. Please cancel claims 12-15, and please amend claim 18 as indicated and move the canceled language as shown from claim 18 to new claims 33-34. Upon entry of these amendments, claims 1-11 and claims 16-34 will be pending in the application. Claims 1, 6, 11, 18, 21, 22, 25, 28, and 29 are independent.

Patentability Over “International Layout in CSS” and “Mastering Excel 97”

The Office has asserted a rejection of claims 1-32 as obvious over “International Layout in CSS”, 01/1999, W2C, pages 1-40 (herein after “CSS”) in view of Chester et al., “Mastering Excel 97”, 1997, Fourth Edition, Sybex, Inc., pages 129-134 (herein after “Excel”). Applicants respectfully traverse.

Claim 11

The proposed CSS-Excel combination fails to teach or suggest “summing the areas of a minimum logical height rectangle with a maximum logical height rectangle, and dividing that sum by twice a physical width proposed by a table sizing method.” Specifically, amended claim 11 recites:

11. (currently amended) A method for auto-sizing textual mark-up language table cells with a vertical character flow property, the method comprising:

determining that a table cell has the vertical character flow property;
estimating a logical width for input to a table cell sizing method designed to auto-size a table cell with the horizontal character flow property; and

calling the table cell sizing method with the estimated logical width input in order to auto-size the table cell with the vertical character flow property;

wherein the estimated logical width is determined by,

dividing the area of a minimum logical height rectangle by a physical width proposed by a table sizing method,

dividing the area of a maximum logical height rectangle by a physical width proposed by a table sizing method, or

summing the areas of a minimum logical height rectangle with a maximum logical height rectangle, and dividing that sum by twice a physical width proposed by a table sizing method.

Applicants respectfully submit that the Office has failed to carry the burden of establishing obviousness because the references provided by the Office fail to teach or suggest “summing the areas of a minimum logical height rectangle with a maximum logical height rectangle, and dividing that sum by twice a physical width proposed by a table sizing method.”

For example, the Action at page 8, states that this arrangement (formerly of claim 15) is described by CSS (“CSS on page 15 teaches determining logical width”). Applicants respectfully agree that CSS at page 15 states the words “logical width.” However, “logical width” is not the recited language. There is simply nothing in CSS alone, or in the proposed CSS-Excel combination, that would lead anyone to the recited arrangement. It appears that this language was simply overlooked by the Office in both the Action mailed July 8, 2004, and in the final Action mailed March 10, 2005.

For example, CSS states nothing about (1) a “rectangle”, (2) a “minimum ... height rectangle”, (3) a “maximum ... height rectangle”, (4) a “minimum logical height rectangle”, (5) “a maximum logical height rectangle”, (6) “summing the areas” of any rectangles, (7) “dividing that sum by twice a physical width proposed”, or (5) “a table sizing method.” To establish a prima facie case of obviousness, the Office must direct Applicants to references that teach or suggest all of a claim’s limitations. There is simply nothing presently in the record discussing these features. The rejection is improper on its face.

For at least this reason claim 11 is in condition for allowance. Such action is respectfully requested.

Claims 16-17

Claims 16-17 depend from claim 11. Since they depend from claim 11, they should be allowed for at least the reasons stated for claim 11. In view of the foregoing discussion of claim 11, the merits of the separate patentability of dependent claims 16-17 are not belabored at this time. Claims 16-17 should be allowable. Such action is respectfully requested.

Claim 18

The proposed CSS-Excel combination fails to teach or suggest “determining a proposed logical width by summing the areas of the determined maximum logical height rectangle and the determined minimum logical height rectangle and dividing the sum by approximately twice the proposed physical width.” This should be apparent in view of the discussion of claim 11. For at least this reason, claim 18 is now in condition for allowance. Such action is respectfully requested.

Claims 19-20

Claims 19-20 depend from claim 18. Since they depend from claim 18, they should be allowed for at least the reasons stated for claim 18. In view of the foregoing reason for allowance of claim 18, the merits of the separate patentability of dependent claims 19-20 are not belabored at this time. Claims 19-20 should be allowable. Such action is respectfully requested.

Claim 21

The proposed CSS-Excel combination fails to teach or suggest “a third program module designed to alter the request made to the second program module by the first program module so that while the second program module continues to operate as it would for a textual mark-up language element with the horizontal character flow property, the second program module is actually auto-sizing a textual mark-up element with the vertical character flow property.”

For example, the Action at page 21, states that this arrangement “incorporates similar subject matter as claimed in claims 1-20, and 26.” The Office states that claims 1-20 and 26 are discussed, for example, in CSS at pages 13-15 and in Excel at pages 129-133. Applicants respectfully disagree. There is nothing in CSS or Excel that describes the recited arrangement.

There is simply no discussion of (1) “a third program module” (2) “designed to alter the request made to the second program module” (3) “by the first program module” (4) “so that while the second program module continues to operate as it would for a textual mark-up language element with the horizontal character flow property,” (5) “the second program module is actually auto-sizing a textual mark-up element with the vertical character flow property.”

For example, “CSS is a style sheets language that can be applied to HTML to control the style of a document: which fonts and colors to use, how much white space to insert, etc.” CSS at page 2. As such, CSS “introduces a number of properties to represent such formatting” such as “vertical layout flow and layout grid.” CSS at page 3. For example, a property “sets the layout flow for the element” (CSS at page 5) and indicates to what elements a property may apply (“Applies to: all elements”, CSS page 4). As is familiar to those of ordinary skill in the art, an element such as a “SPAN” element may have a property, such as “horizontal”, assigned thereto (CSS at page 7 “1996”). However, neither these elements nor the properties assigned thereto would be confused with program modules that read, measure, or display the elements according to the properties. Specifically, there is nothing in CSS that describes (1) “a third program module” (2) “designed to alter the request made to the second program module” (3) “by the first program module” (4) “so that while the second program module continues to operate as it would for a textual mark-up language element with the horizontal character flow property,” (5) “the second program module is actually auto-sizing a textual mark-up element with the vertical character flow property.”

Similarly, Excel fails to teach or suggest the recited arrangement. The Office admits that CSS fails to teach or suggest various features (“auto-sizing” or “table cell sizing”). See e.g., Office Action at page 3, mailed March 10, 2005. However, the Office then asserts that these features are described in Excel at page 129-134. Applicants respectfully disagree.

For example, Excel provides no discussion of (1) “a third program module” (2) “designed to alter the request made to the second program module” (3) “by the first program module” (4) “so that while the second program module continues to operate as it would for a textual mark-up language element with the horizontal character flow property,” (5) “the second program module is actually auto-sizing a textual mark-up element with the vertical character flow property.”

The Office appears to be relying on the display of vertical text on screen (see e.g., Excel at page 134), as proof that vertical text was prior art. Applicants agree that Excel at page 134 depicts vertical text. Additionally, Excel depicts a graphical user interface that provides “horizontal and vertical alignment of text in selected cells.” Excel at page 129. However, the recited language is not so broad to claim “vertical text.”

Nothing in Excel would lead one of ordinary skill in the art to the recited language--(1) “a third program module” (2) “designed to alter the request made to the second program module” (3) “by the first program module” (4) “so that while the second program module continues to operate as it would for a textual mark-up language element with the horizontal character flow property,” (5) “the second program module is actually auto-sizing a textual mark-up element with the vertical character flow property.” To establish a prima facie case of obviousness, the Office must direct Applicants to references that teach or suggest all of a claim’s limitations. There is simply nothing presently in the proposed CSS-Excel combination discussing these features.

For at least this reason claim 21 is in condition for allowance. Such action is respectfully requested.

Claim 22

The proposed CSS-Excel combination fails to teach or suggest “wherein the first module is designed to request auto-sizing of table cell elements with the horizontal character flow property and the third module alters inputs of the first module’s requests to the second module when the request is to auto-size a table cell element with the vertical character flow property.” This should be apparent in view of the discussion of claim 21. For at least this reason, claim 22 is now in condition for allowance. Such action is respectfully requested.

Claims 23-24

Claims 23-24 depend from claim 22. Since they depend from claim 22, they should be allowed for at least the reasons stated for claim 22. In view of the foregoing reason for allowance of claim 22, the merits of the separate patentability of dependent claims 23-24 are not belabored at this time. Claims 23-24 should be allowable. Such action is respectfully requested.

Claim 25

The proposed CSS-Excel combination fails to teach or suggest “wherein the first module is designed to request auto-sizing of table cell elements with the horizontal character flow

property and the third module alters inputs of the first module's requests to the second module when the request is to auto-size a table cell element with the vertical character flow property." This should be apparent in view of the discussion of claim 1. For at least this reason, claim 25 is now in condition for allowance. Such action is respectfully requested.

Claims 26-27

Claims 26-27 depend from claim 25. Since they depend from claim 25, they should be allowed for at least the reasons stated for claim 25. In view of the foregoing reason for allowance of claim 25, the merits of the separate patentability of dependent claims 26-27 are not belabored at this time. Claims 26-27 should be allowable. Such action is respectfully requested.

Claim 28

The proposed CSS-Excel combination fails to teach or suggest "a third object designed to alter the request made to the second object by the first object so that while the second object continues to operate as it would for a textual mark-up language element with the horizontal character flow property, the second object is actually auto-sizing a textual mark-up element with the vertical character flow property and the resulting auto-sized element is returned to the first object rotated ninety degrees clockwise."

For example, "CSS is a style sheets language that can be applied to HTML to control the style of a document: which fonts and colors to use, how much white space to insert, etc." CSS at page 2. As such, CSS "introduces a number of properties to represent such formatting" such as "vertical layout flow and layout grid." CSS at page 3. For example, a property "sets the layout flow for the element" (CSS at page 5) and indicates to what elements a property may apply ("Applies to: all elements", CSS page 4). As is familiar to those of ordinary skill in the art, an element such as a "SPAN" element may have a property, such as "horizontal", assigned thereto (CSS at page 7 "1996"). However, neither these elements nor the properties assigned thereto would be confused with program objects that manipulate the elements according to the properties. Specifically, there is nothing in CSS that describes (1) "a third object designed to alter the request made to the second object by the first

object” (2) “so that while the second object continues to operate as it would for a textual mark-up language element with the horizontal character flow property, the second object is actually auto-sizing a textual mark-up element with the vertical character flow property” and (3) “the resulting auto-sized element is returned to the first object rotated ninety degrees clockwise.”

Similarly, Excel fails to teach or suggest the recited arrangement. The Office admits that CSS fails to teach or suggest various features (“auto-sizing” or “table cell sizing”). See e.g., Office Action at page 3, mailed March 10, 2005. However, the Office then asserts that these features are described in Excel at page 129-134. Applicants respectfully disagree.

For example, Excel provides no discussion of (1) “a third object designed to alter the request made to the second object by the first object” (2) “so that while the second object continues to operate as it would for a textual mark-up language element with the horizontal character flow property, the second object is actually auto-sizing a textual mark-up element with the vertical character flow property” and (3) “the resulting auto-sized element is returned to the first object rotated ninety degrees clockwise.”

The Office appears to be relying on the display of vertical text on screen (see e.g., Excel at page 134), as proof that vertical text was prior art. Applicants agree that Excel at page 134 depicts vertical text. Additionally, Excel depicts a graphical user interface that provides “horizontal and vertical alignment of text in selected cells.” Excel at page 129. However, the recited language is not so broad to claim “vertical text.”

Nothing in Excel would lead one of ordinary skill in the art to the recited language--(1) “a third object designed to alter the request made to the second object by the first object” (2) “so that while the second object continues to operate as it would for a textual mark-up language element with the horizontal character flow property, the second object is actually auto-sizing a textual mark-up element with the vertical character flow property” and (3) “the resulting auto-sized element is returned to the first object rotated ninety degrees clockwise.” To establish a prima facie case of obviousness, the Office must direct Applicants to references that teach or suggest all of a claim’s limitations. There is simply nothing presently in the proposed CSS-Excel combination discussing these features.

For at least this reason claim 28 is in condition for allowance. Such action is respectfully requested.

Claim 29

The proposed CSS-Excel combination fails to teach or suggest “the second object returns the resulting height and width measurements of the table cell element to the third object, the returned width measured in the direction parallel to the character flow direction, and the returned height measured in the direction perpendicular to the character flow direction, and the third object forwards the returned height measurement to the first object as the measurement for the maximum table cell width.”

For example, “CSS is a style sheets language that can be applied to HTML to control the style of a document: which fonts and colors to use, how much white space to insert, etc.” CSS at page 2. As such, CSS “introduces a number of properties to represent such formatting” such as “vertical layout flow and layout grid.” CSS at page 3. For example, a property “sets the layout flow for the element” (CSS at page 5) and indicates to what elements a property may apply (“Applies to: all elements”, CSS page 4). As is familiar to those of ordinary skill in the art, an element such as a “SPAN” element may have a property, such as “horizontal”, assigned thereto (CSS at page 7 “1996”). However, neither these elements nor the properties assigned thereto would be confused with program modules that manipulate the elements according to the properties. Specifically, there is nothing in CSS that describes (1) “the second object returns the resulting height and width measurements of the table cell element to the third object,” (2) “the returned width measured in the direction parallel to the character flow direction, and the returned height measured in the direction perpendicular to the character flow direction,” and (3) “the third object forwards the returned height measurement to the first object as the measurement for the maximum table cell width.”

Similarly, Excel fails to teach or suggest the recited arrangement. The Office admits that CSS fails to teach or suggest various features (“auto-sizing” or “table cell sizing”). See e.g., Office Action at page 3, mailed March 10, 2005. However, the Office then asserts that these features are described in Excel at page 129-134. Applicants respectfully disagree.

For example, Excel provides no discussion of (1) “the second object returns the resulting height and width measurements of the table cell element to the third object,” (2) “the returned width measured in the direction parallel to the character flow direction, and the returned height measured in the direction perpendicular to the character flow direction,” and (3) “the third object forwards the returned height measurement to the first object as the measurement for the maximum table cell width.”

The Office appears to be relying on the display of vertical text on screen (see e.g., Excel at page 134), as proof that vertical text was prior art. Applicants agree that Excel at page 134 depicts vertical text. Additionally, Excel depicts a graphical user interface that provides “horizontal and vertical alignment of text in selected cells.” Excel at page 129. However, the recited language is not so broad to claim “vertical text.”

Nothing in Excel would lead one of ordinary skill in the art to the recited language--(1) “the second object returns the resulting height and width measurements of the table cell element to the third object,” (2) “the returned width measured in the direction parallel to the character flow direction, and the returned height measured in the direction perpendicular to the character flow direction,” and (3) “the third object forwards the returned height measurement to the first object as the measurement for the maximum table cell width.” To establish a prima facie case of obviousness, the Office must direct Applicants to references that teach or suggest all of a claim’s limitations. There is simply nothing presently in the proposed CSS-Excel combination discussing these features.

For at least this reason claim 29 is in condition for allowance. Such action is respectfully requested.

Claims 30-32

Claims 30-32 depend from claim 25. Since they depend from claim 25, they should be allowed for at least the reasons stated for claim 25. In view of the foregoing reason for allowance of claim 25, the merits of the separate patentability of dependent claims 30-32 are not belabored at this time. Claims 30-32 should be allowable. Such action is respectfully requested.

Claim 1

The proposed CSS-Excel combination fails to teach or suggest “in the case of determining that a table sizing code segment is causing a table cell sizing code segment to execute a code path determining a minimum width table cell measurement for a given table cell element with a vertical character flow property, causing the table cell sizing code segment to execute using a maximum width input that is likely to cause element content to be flowed into a single vertical line.”

For example, “CSS is a style sheets language that can be applied to HTML to control the style of a document: which fonts and colors to use, how much white space to insert, etc.” CSS at page 2. As such, CSS “introduces a number of properties to represent such formatting” such as “vertical layout flow and layout grid.” CSS at page 3. For example, a property “sets the layout flow for the element” (CSS at page 5) and indicates to what elements a property may apply (“Applies to: all elements”, CSS page 4). As is familiar to those of ordinary skill in the art, an element such as a “SPAN” element may have a property, such as “horizontal”, assigned thereto (CSS at page 7 “1996”). However, neither these elements nor the properties assigned thereto would be confused with code segments that measure or size the elements according to the properties. Specifically, there is nothing in CSS that describes (1) “in the case of determining that a table sizing code segment is causing a table cell sizing code segment” (2) “to execute a code path determining a minimum width table cell measurement for a given table cell element with a vertical character flow property,” (3) “causing the table cell sizing code segment to execute using a maximum width input that is likely to cause element content to be flowed into a single vertical line.”

Similarly, Excel fails to teach or suggest the recited arrangement. The Office admits that CSS fails to teach or suggest various features (“auto-sizing” or “table cell sizing”). See e.g., Office Action at page 3, mailed March 10, 2005. However, the Office then asserts that these features are described in Excel at page 129-134. Applicants respectfully disagree.

For example, Excel provides no discussion of (1) “in the case of determining that a table sizing code segment is causing a table cell sizing code segment” (2) “to execute a code path determining a minimum width table cell measurement for a given table cell element with a

vertical character flow property,” (3) “causing the table cell sizing code segment to execute using a maximum width input that is likely to cause element content to be flowed into a single vertical line.”

The Office appears to be relying on the display of vertical text on screen (see e.g., Excel at page 134), as proof that vertical text was prior art. Applicants agree that Excel at page 134 depicts vertical text. Additionally, Excel depicts a graphical user interface that provides “horizontal and vertical alignment of text in selected cells.” Excel at page 129. However, the recited language is not so broad to claim “vertical text.”

Nothing in Excel would lead one of ordinary skill in the art to the recited language--(1) “in the case of determining that a table sizing code segment is causing a table cell sizing code segment” (2) “to execute a code path determining a minimum width table cell measurement for a given table cell element with a vertical character flow property,” (3) “causing the table cell sizing code segment to execute using a maximum width input that is likely to cause element content to be flowed into a single vertical line.” To establish a prima facie case of obviousness, the Office must direct Applicants to references that teach or suggest all of a claim’s limitations. There is simply nothing presently in the proposed CSS-Excel combination discussing these features.

For at least this reason claim 1 is in condition for allowance. Such action is respectfully requested.

Claims 2-5

Claims 2-5 depend from claim 1. Since they depend from claim 1, they should be allowed for at least the reasons stated for claim 1. In view of the foregoing discussion of claim 1, the merits of the separate patentability of dependent claims 2-5 are not belabored at this time. Claims 2-5 should be allowable. Such action is respectfully requested.

Claim 6

The proposed CSS-Excel combination fails to teach or suggest “determining a maximum distance that characters will be allowed to flow in the vertical character flow direction, the

maximum distance determination being made based on some empirically determined number N multiplied by the average character logical width.”

For example, the Action at page 6, states that “CSS on pages 13-15” describes the recited arrangement. However, Applicants disagree. For example, CSS states nothing about a “maximum distance determination” based on a “number N multiplied by the average character width.” Further, CSS fails to discuss either an “empirically determined number N” or an “average character ... width.”

CSS at 13-15 mentions (1) a “largest character”, (2) “layout flow is vertical”, (3) a “widest character”, and (4) an “author preferred specific number of characters ... in a line.” However, a “widest character” or a “largest character” teaches away from an “average character ... width”, and “an authored preferred ... number of characters” teaches away from an “empirically determine number N.” The proposed CSS-Excel combination simply fails to teach or suggest “determining a maximum distance that characters will be allowed to flow in the vertical character flow direction, the maximum distance determination being made based on some empirically determined number N multiplied by the average character logical width”.

For at least this reason amended claim 6 is in condition for allowance. Such action is respectfully requested.

Claims 7-10

Claims 7-10 depend from claim 6. Since they depend from claim 6, they should be allowed for at least the reasons stated for claim 6. In view of the foregoing discussion of claim 6, the merits of the separate patentability of dependent claims 7-10 are not belabored at this time. Claims 7-10 should be allowable. Such action is respectfully requested.

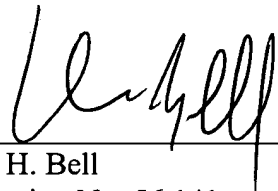
Conclusion

The claims in their present form should now be allowable. Such action is respectfully requested.

Respectfully submitted,

KLARQUIST SPARKMAN, LLP

By



Daniel H. Bell
Registration No. 56,141

One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Telephone: (503) 595-5300
Facsimile: (503) 228-9446